Exhibit C

ebay™ Billpoint

Patent Proposal

Tracking No: [to be completed by Patent Coordinator] _____

Date: _____ (Billpoint)

Title: Fraud Detection

Product(s): _____

Date first disclosed outside the company or offered for sale: Internal — aren't live until late May/June 1999 — no public disclosure

Inventor(s): Jason May, Ian Flint, others

Description: *Please provide a short description of the invention*

The system will analyze incoming transactions for potential fraud by applying a set of rules that recognize unusual behavior. Transactions that violate one or more rules will be assigned a score; transactions with high scores will be flagged as potentially fraudulent. Such transactions may be routed to a human operator for investigation, or may be automatically handled by the system (such handling might involve outright rejection or trigger electronic requests for further information from the customer).

The fraud analysis rules may include the following:
"Suspect data checks": Checks that recognize suspicious data values. e.g. use of post office boxes for shipping, use of different billing and shipping addresses, use of credit cards from financial institutions with unreliable security records.

"Velocity checks": Checks that look for unusual amounts of behavior over a period of time. e.g. usage of the same shipping address on multiple transactions

Read and Understood:_____

Dated:_____

Confidential when filled in

OVERVIEW

The Billpoint system is a service for enabling payment transactions between buyers and sellers over the Internet. Billpoint is a consumer service, and it is essential that the system ultimately scale to support a high volume of payment transactions between potentially millions of buyers and sellers.

Billpoint provides a full suite of services to facilitate payment transactions, acting as a trusted intermediary during the transaction. The primary target customer base is individuals or small businesses, who could not otherwise afford to build or buy the infrastructure to accept credit cards for Internet purchases.

Feature Set

Flexible Product Support

Physical goods

Services

Digital content

Multiple-item purchases

Seller Product Administration

Hosting of product information

Unlimited dynamic product definition driven by seller's Web site or database

Hosting of offer and acceptance within Billpoint for one-time exchanges

Flexible Payment Methods

Credit cards

Checks or money orders

Wire transfers

Manual entry of order by seller (telephone, fax or e-mail orders)

Flexible Payment Method Support

Payment in advance, on fulfillment, or on receipt

Aggregated payments for digital content

Subscriptions and bundles

Recurring charges (for ongoing services)

Partial fulfillment of multiple-item orders

Account Management

Linked buyer accounts for families

Linked buyer accounts for corporations (e.g. corporate card purchasing)

Purchase access control by seller or seller category (e.g. lockouts for children)

Single user account managing multiple seller accounts

Fl xible F e Structures

Variable per seller

Seller categories

Volume-based discounts

Customer Service

Electronic mail automation

Account Access

Buyer activity reporting

Seller activity reporting

Fraud Detection and Prevention

Real time — postal address and telephone number validation

Delayed — analysis of transaction patterns

On-line dispute handling

Suspect-monitoring utilities

Data Analysis

Transaction pattern analysis (see Fraud Prevention)

Management reporting

Joint marketing campaigns

Partner Support

Integrated seller registration for malls and portals

Special fee discount structures

Revenue splits

Front-end Integration

Seamless integration with marketplace systems

Pass through buyer registration information from seller's site

Integration with e-commerce software

Back-end Integration

Address and telephone number validation (see Fraud Prevention)

Sales tax calculations

Shipping cost calculations

Shipment status integration (e.g. SmartShip, iShip)

Customer Servic

Reporting tools

Seller activation and feature access control

Transaction refund and cancellation utilities

Architecture Highlights

- No direct connection required between merchant or marketplace server and Billpoint
- Traffic through buyer's browser can be digitally signed for security
- Control over what buyer information is disclosed to the seller during a transaction
- N-tier architecture, with persistent application servers
- Servers are stateless and can be distributed across multiple machines
- Separate transaction and analysis databases
- Event-driven transaction model for modular expansion

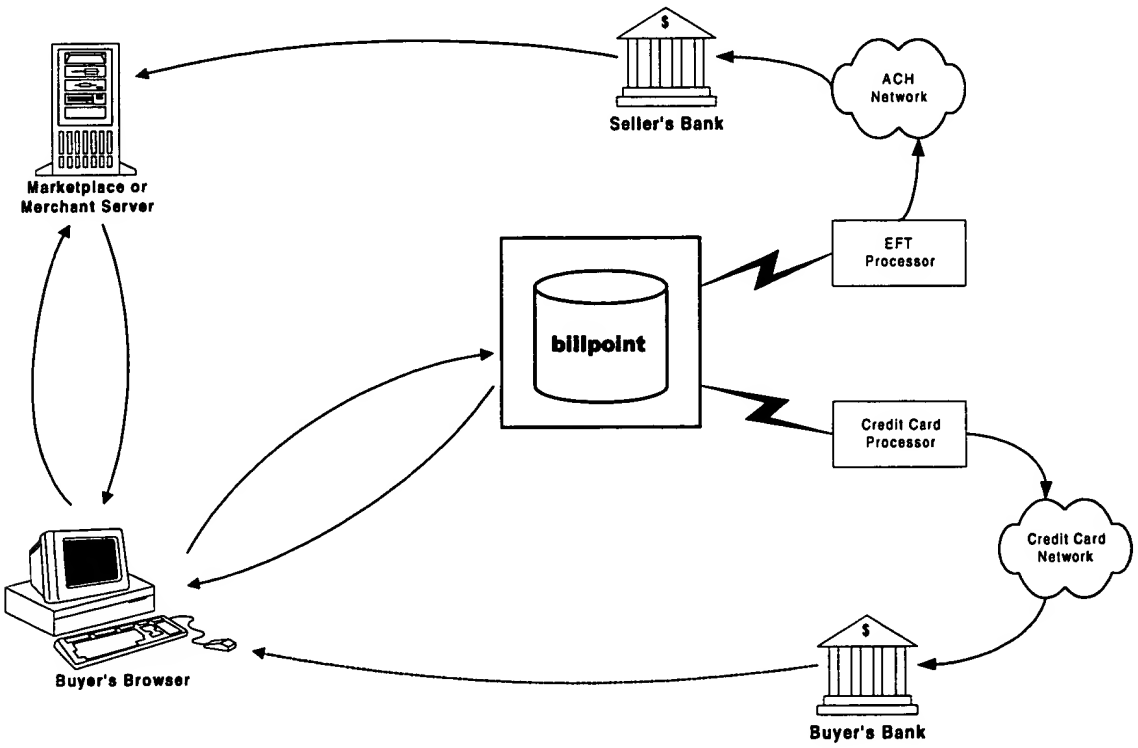
SIMPLICITY

From an end user (buyer or seller) perspective, the Billpoint service attempts to be as simple and straightforward as possible. An easy-to-understand interface enables customer transactions to be conducted more efficiently, reducing the load on the system as well as reducing requirements for human customer service assistance.

Key aspects of the Billpoint functionality where simplicity is essential include:

- Seller Registration and Activation
- Integration with Marketplace (auctions, classifieds, communities) systems, including seamless registration
- Seller Product Configuration
- Transaction Execution and Order Fulfillment
- Buyer and Seller Account Access, including Dispute Resolution

Many of these processes can be implemented as sequential request-response process flows.



Marketplace Integration

Billpoint integration with marketplace and merchant systems is purely a "front-end" integration. Software requirements on the marketplace server are limited to CGI scripts (or servlets, or other server-side code) that exchange information between Billpoint and the marketplace system through HTTP FORM POST operations. No direct database integration is required.

Information exchange between merchant server and Billpoint can be enabled for the following:

- Seller account registration
- Buyer account registration
- Buyer address data pass-through

Customer Assistance

On-line assistance will be provided throughout, in the form of:

- Extensive seller registration assistance (FAQ lists, configuration automation)
- Context-sensitive help during account management and transactions
- Tools available to Billpoint customer service staff for rapid disposition of customer inquiries

STABILITY

The Billpoint system is intended to support a large number of buyers and sellers who are distributed across the globe, executing transactions at any time of day. Since the ability to execute transactions may be business-critical to these customers, Billpoint will guarantee an extremely high level of system availability. To provide this level of service, several areas must be addressed:

- Stability of physical environment
- Stable application architecture
- Change management
- Monitoring facilities

It should be noted that the Billpoint system is intended to increase customer and transaction volume steadily over time. The stability and scalability requirements described below are critical for the long term goal, but while volume remains low these requirements can be greatly relaxed.

Physical Environment

Proven Technology

Billpoint will employ only known, tested, stable infrastructure technology. Beta products and bleeding-edge technology will be avoided. The preliminary platform is as follows:

- Sun SPARC servers, running current mainstream version of Solaris. Database an application servers will initially be 4500-class machines; Web servers will initially be Ultra 5s.
- Oracle 8 database
- Apache-Stronghold Web server
- Proven commercial middleware technology: Tibco Rendezvous.

Production server clusters will be hosted at a high-end commercial hosting service provider (Exodus Communications).

Data Resilience

Data mirroring and RAID are required for all storage devices containing Billpoint customer and transaction data.

Multiple Data Centers

Billpoint transaction-processing server clusters can be installed at several different geographical locations. If a single data center location goes down, transaction volume can be shunted seamlessly to the remaining locations.

Third-party technology such as Cisco LoadDirector or Resonate Dispatch will be employed to provide this transparency.

Server Fault Tolerance

Each server cluster could consist of multiple machines with redundant services.

Multiple connections to credit-card proc ssing s rvic s

Each data center will have at least one external connection for credit card authorization.

Credit card processing connections will be established through multiple vendors to maintain availability in case of systems failure on the vendor's side. Quality-of-service guarantees will be required of vendors.

Application Architecture

Data Replication

To support databases hosted at multiple data centers as described above, shared Billpoint data (such as buyer and seller profile information) will be replicated in close-to-real-time to all the data centers.

Replication of transaction data is not subject to as strict a requirement as the shared data. A full transaction record must be able to be constructed when a customer requests an on-line account statement, but this can be achieved in a variety of ways, not just replication.

Data Warehouse

A core database design principle is that reporting and analysis facilities should never be executed against the same database as live business transactions. Analysis functions typically take a much longer time to execute than transaction activity, and conflicts between these application loads can greatly reduce customer perceived performance.

Billpoint reporting and analysis functions will all be carried out against a separate data warehouse system, and will have no impact on customer transaction activity.

Application Server Redundancy

Multiple instances of each application server can be run in parallel to balance the system load appropriately between the different functions. Applications servers can be installed on different physical machines.

In the n-tier environment, the transaction management layer will balance the application load transparently across the servers.

Multiple instances of each application server can be run in parallel to balance the system load appropriately between the different functions. Applications servers can be installed on different physical machines.

Component Resilience

Application servers must be able to handle database downtime cleanly:

- Database not reachable at server startup—send alert message, print error on STDERR, and abort
- Database query fails due to Oracle access error—send alert message, print error on STDERR, abort transaction, clean up residual connection
- Database reconnect fails—send alert message, print error on STDERR, abort transaction

Application servers should continue to reattempt database connection following an outage.

Change Management

Zero-down-time release control

New functionality must be able to be released to the Billpoint production environment without requiring any downtime that is perceptible to the end user. The server architecture is designed to permit for rapid replacement of individual components.

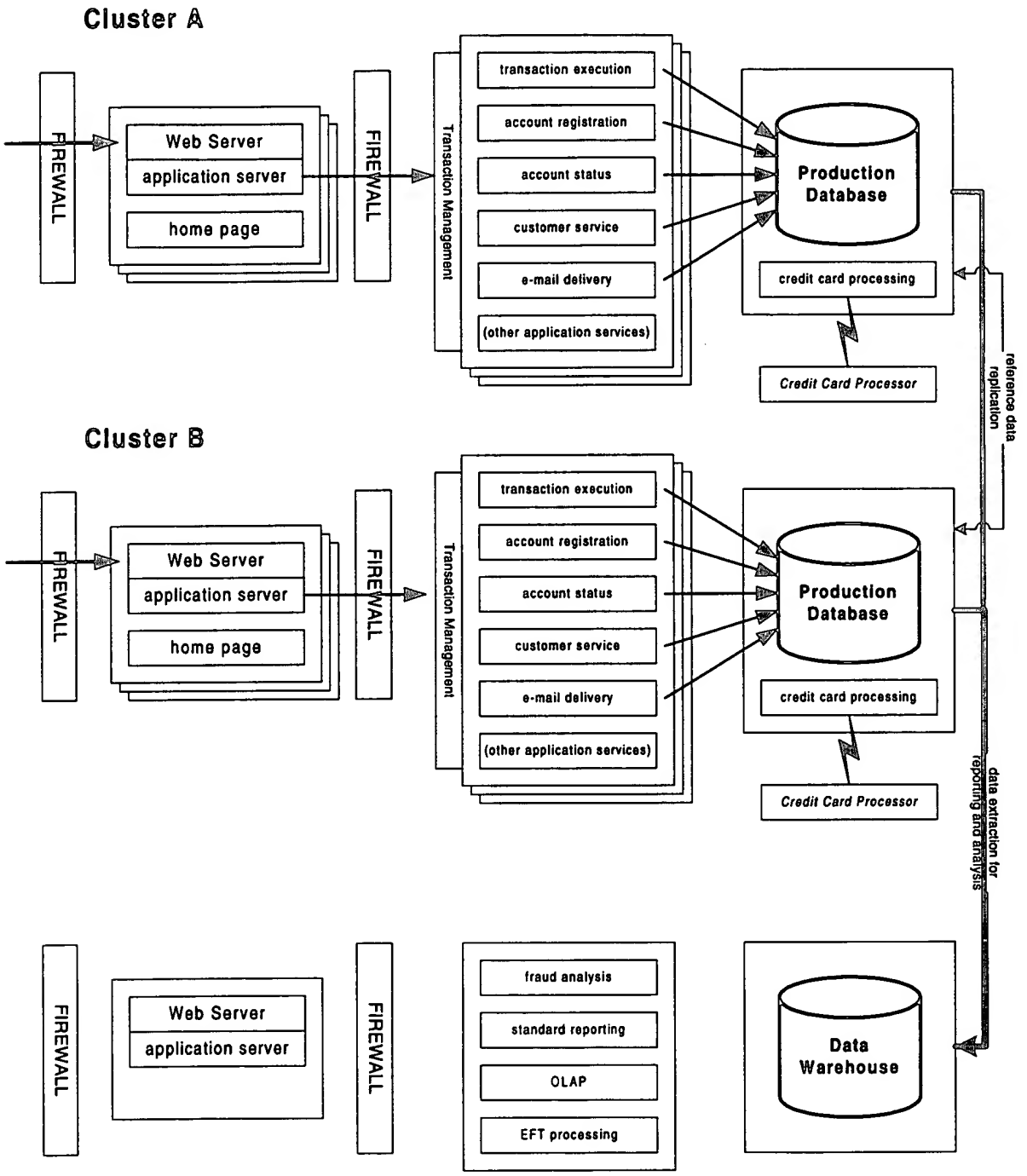
Each release will be strictly backwards-compatible with the previous release.

In particular, database changes and application functionality changes will generally need to be released incrementally. Dependencies between changes in different application components are to be avoided.

Monitoring Facilities

Real-time monitoring of application functions

Immediate response by support staff is essential. Monitoring systems will provide real-time visibility to server activity at all times, including software-generated alerts via e-mail and pager where necessary.



Warehouse Cluster (Access limited to Billpoint staff and authorized marketplace partners)

SCALABILITY

The transaction-processing infrastructure is able to scale smoothly to support a large volume of transactions. Aspects of scalability can be divided into two major areas:

- Hardware & network infrastructure scalability — As hardware or network capacity is added to the system it should be able to take advantage of the new capacity in as linear a fashion as possible.
- Software architecture scalability — There should be no inherent bottlenecks that will require major re-engineering. Any performance bottlenecks that are introduced in the course of business decisions driving implementation (e.g. adding new features to the system) must be obvious and any capacity limitations that are created should be able to be remedied in the future.

It should be noted that the Billpoint system is intended to increase customer and transaction volume over time. While the initial capacity figures (one million purchase transactions per month) may be over-ambitious in the short-term (6-9 months), over the longer term the continued success of the Internet is likely to drive much higher e-commerce transaction volume that this. Billpoint must be able to scale to support even the most optimistic estimates for Internet purchase transaction activity.

Infrastructure Scalability

The Billpoint software takes advantage of the scalability available through the server hardware and operating system and the application database.

Capacity estimate

Assumption: 1 million Billpoint transactions per month, occurring during a 6-hour window

Each transaction could potentially map to (high estimate):

- 60 SQL queries (single-row selects, inserts and updates)
- 40 page views
- 10 email messages
- 2 credit card authorizations

Converted to expected peak load per second:

- 90 SQL queries
- 60 page views
- 15 email messages
- 3 credit card authorizations

Prior experience has shown that the database load can be accommodated with ease by the target hardware platform, and the network traffic from the Web page and email delivery is also well within normal network capacity.

Data Architecture Scalability

As volume grows, an analytical capability will be required for fraud prevention and for data mining. This environment will be hosted independently from the transaction database(s), to avoid performance degradation from mixed workloads.

The transaction databases will be tuned for end-user performance (single inserts during transactions and range selects for account access).

The data warehouse environment will be centralized and tuned appropriately for Billpoint analysis requirements.

Process Automation

Manual processes must be eliminated or minimized wherever possible:

- Seller registration process is automated, with "wizards" to aid sellers in configuration tasks.
- Outbound email delivery will be entirely automated.
- Email responses will pass through an automated handling process, which will recognize standard error formats and deal with invalid email addresses (a potential fraud indicator).

Software Architecture Scalability

The Billpoint software is divided into several independent server components.

Application servers are persistent, eliminating process launch overhead.

Database connections are persistent and reused between transactions.

Transaction volume is load-shared across servers of the same type.

The servers are stateless, and can be distributed across multiple server machines. Different transaction servers can execute different steps of the same transaction.

Process architecture based on an event-driven (publish/subscribe) model with a high degree of granularity so that new features can be introduced quickly.

SECURITY

There are two primary drivers for security services within the Billpoint architecture:

- Protection of confidential customer information
- Integrity of Billpoint transactions

Information Asset Protection

The Billpoint database must be protected from unauthorized access at all times. In particular, the customer credit card repository. The following measures will be taken to defend the database:

- Physical security — provided by the data center.
- Network security — dual firewall isolating the database from the Internet. The only access permitted to the database will be from the Billpoint application servers, making valid Billpoint database requests.

Buyer Privacy Protection

The Billpoint system will never release information about buyers unless explicit permission has been given to do so. During a purchase transaction, whenever the seller requires buyer information to complete a transaction, the buyer will be explicitly informed of what information will be disclosed.

Transaction Integrity

All data communications between the buyer's browser and the Billpoint Web server will be encrypted using the industry-standard SSL protocol. Billpoint will not support browsers that do not support SSL.

Credit card information is only entered when the buyer originally opens an account with Billpoint. This information is never disclosed to the seller and is therefore never transmitted outside of the Billpoint system. This eliminates one potential area of security concern.

Digital Signatures

When a transaction is taking place, there is the potential for the data payload (HTTP FORM parameters) to be altered in transit prior to the SSL encryption stage. Such tampering could be carried out be the buyer or by an unauthorized third party.

Such tampering is of greatest concern for digital content, where the objective of tampering would be to obtain access to the seller's product without paying the amount that is charged.

Billpoint prevents such tampering by implementing a digital signature on the data payload that is transmitted from the seller's server to Billpoint. Use of the digital signature is at the seller's option, and should only be used when there is a need to prevent tampering.

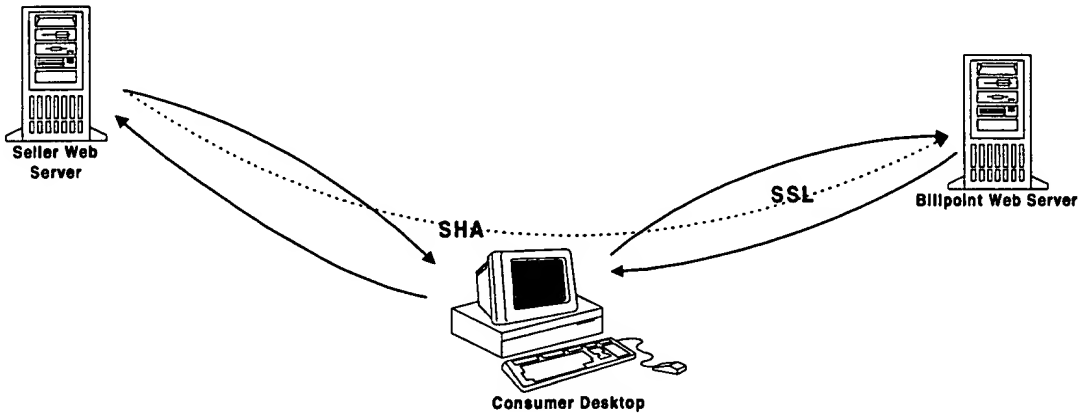
Billpoint utilizes the Secure Hash Algorithm (SHA) for generating digital signatures. All the message parameters are included in the SHA digest, along with a secret password that is not included in the message. The password shared between the seller and Billpoint, and is generated during the registration process.

The digital signature security measures can be compromised if an authorized third party obtains the seller's password by breaking the security of the seller's own system, for which Billpoint is not responsible.

Transaction Records

Billpoint retains a complete record of every transaction that takes place. This history will be kept for as long as there is any possibility of fraud or other concern.

Buyers and sellers will be provided with access to their own transaction records. Individual Billpoint transaction records will not be disclosed to any other party.



FEATURES

Seller Product Options

Physical Goods

- Require shipment by the seller
- Seller must explicitly fulfill the order when shipping
- Buyer credit card is not charged until order is fulfilled
- If seller cancels the order (for any reason), credit card is not charged and Billpoint charges no fee
- If buyer credit card is invalid at time of fulfillment, seller is notified instantly and fulfillment is not completed within Billpoint
- Billpoint remits funds to seller when buyer has confirmed receipt of goods or a time limit expires (default: 30 days)
- Seller may require shipping address information from buyer through disclosure mechanism

Services

- Buyer credit card is charged immediately at time of purchase
- If buyer credit card is invalid, purchase will not be completed (no Billpoint transaction id assigned)
- Billpoint remits funds to seller in next payment cycle
- Seller may require address information from buyer through disclosure mechanism

Digital Goods

- Currently not available

Shipment Tracking

Carriers

Federal Express

UPS

United States Postal Service

Airborne Express

DHL

Access Mechanism

HTML scraping from carrier Web site package tracking pages

Dedicated servers will download package status on request, given a list of package identifiers

Local Storage

Record package status information in Billpoint database: package id (index), date last checked, date delivered (NULL until delivered), full HTML text of package status from carrier's site.

Application servers will access status information from the database, not from the tracking servers

Exc ption Handling

Tracking process must handle:

- Package number rejected by carrier's system
- Package number does not exist (has not been shipped yet)
- Carrier Web site not available
- Parse failure (carrier modified Web site)

Architecture

Dedicated server for each carrier

Server listens for messages containing package identifiers

Server scrapes carrier Web site and updates database table with status for each package

Control server that dispatches requests to the servers

Cron job that periodically looks up list of package ids to be checks and sends out commands

AUTHORIZATION PROCESS

Credit card authorizations occur in the following situations:

- Purchases for services, at time of purchase
- For physical goods, at time of fulfillment
- During a refund

We do **not** authorize the card in these cases:

- At time of buyer registration
- At time purchase for physical goods

Process flow for authorizations for service purchases:

1. xn_server sends message to auth_broker, starts giveup timer, and returns without sending a response to the buyer's browser
2. auth_broker receives the message and transmits to the payment processor
3. auth_broker receives response from the payment processor, sends message to xn_server (not necessarily the same xn_server that send the original request)
4. xn_server receives the response, recovers the order information, checks whether the giveup timer triggered, updates the database and sends a response back to the buyer's browser

Exception conditions:

When the xn_server giveup timer triggers, it checks the following conditions:

- card auth took place and succeeded, and response was successfully sent back to a xn_server, indicated by a record in the CC_AUTH_TO_TRANSACTION table
- card auth took place and failed, and response was successfully sent back to a xn_server, indicated by the absence of a record in the SESSION_TRANSACTION table

If the giveup timer triggered and then a response comes back from the auth_broker, the buyer has already been told the auth failed, so we should throw away the results.